UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/823,894	03/31/2001	Jochen Kappel	051207-1050	7566

22827 7590 04/21/2005

DORITY & MANNING, P.A.
POST OFFICE BOX 1449
GREENVILLE, SC 29602-1449

EXAMINER	
SALL, EL HADJI MALICK	

ART UNIT	PAPER NUMBER
2157	

DATE MAILED: 04/21/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/823,894	KAPPEL ET AL.	
	Examiner	Art Unit	
	El Hadji M Sall	2157	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *03 January 2005*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-27* is/are pending in the application.

4a) Of the above claim(s) *4,9,14 and 19* is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-3,5-8,10-13,15-18 and 20-29* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

1. ***DETAILED ACTION***

This action is responsive to the amendment files on December 30, 2004. Claims 1-3, 5-8, 10-13, 15-18 and 20-29 are pending. Claims 4, 9, 14 and 19 are cancelled. Claims 1-3, 5-8, 10-13, 15-18 and 20-29 represent Corba Jellybens System and Method.

2. ***Claim Rejections - 35 USC § 112***

The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

Claims 6 and 11 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter, which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention. Applicant states in his arguments that both claims 6 and 11 set forth features regarding the automatic generation of a bean in accordance with aspects of the present subject matter, which can be found in the specification. However, automatically providing at least one property that a bean represents, automatically providing a property field that describes the usage of an

attribute for the bean, and automatically providing a type that describes the bean is not described in the specification.

3. Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) a patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims 1-27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Apte et al. U.S. 6,269,373 in view of Chan et al. U.S. 6,836,889.

Apte teaches the invention substantially as claimed including a method and system for persisting beans as container-managed fields.

As to claim 1, Apte teaches a system for providing interaction between a client and a server, comprising:

means for providing at least one property that a bean represents (column 7, lines 18-22, Apte discloses two java beans may be employed that implement the client object 400 and server object 402. What makes a bean different from a pure object is that it has an external interface, called the properties);

means for providing a property field that describes usage of an attribute for the bean (figure 12; column 16, lines 57-65, Apte discloses ...bean data is automatically maintained by the container using a mechanism of its choosing... when a bean chooses to have its persistence container-managed, it specifies which of its fields are to be retained);

means for providing a type that describes the bean (column 2, lines 49-51, Apte discloses the bean may have container-managed fields of both primitive and complex data-types...); and

means for utilizing the bean as a storage entity (column 16, lines 8-9 and 53-56, Apte discloses an entity bean may be either a bean-managed or container-managed (i.e. where an entity bean is an entity bean is used to represent data, which can be a row or a table in a database) and the bean is entirely responsible for storing and retrieving its instance data).

Apte fails to teach explicitly the bean is utilized as a cache entity.

However, Chan teaches code wrapping to simplify access to and use of enterprise java beans. Chan teaches the bean is utilized as a cache (column 7, lines 52-56, Chan discloses a type 2 access bean has a cache called a CopyHelper).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Apte in view of Chan to provide means for utilizing the bean a cache entity. One would be motivated to do so to allow the speed up of data transfer.

As to claim 2, Apte teaches the system of claim 1, further comprising: means for mapping the at least one property to a method in a stub (column 7, lines 20-24, Apte discloses what makes a bean different from a pure object is that it has an external interface, called the properties interface, which allows a tool to read what the component is supposed to do and hook it up to other beans and plug it into another environment).

As to claim 3, Apte teaches the system of claim 1, further comprising: means for providing a handle for the bean, wherein the handle is used for transaction contact (column 16, lines 50-51, Apte discloses obtaining the bean as a handle; column 15, line 67 to column 16, line 2, and lines 35-38, Apte discloses the container generates a serializable handle class, providing a way to identify a bean (i.e. providing a way to identify a bean where by identifying a bean "transaction contact is performed), and the container-provided identity passes when the session bean is removed by the client at the end of the session).

As to claim 5, Apte teaches the system of claim 1, further comprising: means for indicating if a value of one of the at least one property has changed (column 16, lines 23-27, Apte discloses stateful session beans maintain data consistency by updating their fields each time a transaction is committed while informing of changes).

As to claims 6 and 11, Apte teaches a method and a computer readable medium for providing interaction between a client and a server, comprising:

automatically providing at least one property that a bean represents (column 7, lines 18-22, Apte discloses two java beans may be employed that implement the client object 400 and server object 402. What makes a bean different from a pure object is that it has an external interface, called the properties);

automatically providing a property field that describes usage of an attribute for the bean (figure 12; column 16, lines 57-65, Apte discloses ...bean data is automatically maintained by the container using a mechanism of its choosing... when a bean chooses to have its persistence container-managed, it specifies which of its fields are to be retained);

automatically providing a type that describes the bean (column 2, lines 49-51, Apte discloses the bean may have container-managed fields of both primitive and complex data-types...); and

using the bean to provide interaction between a client and a server (column 3, lines 23-26, Apte discloses in figures 9A-9D a distributed application in which a Java client invoke a method from an Enterprise JavaBean running in a CORBA server).

As to claims 7 and 12, Apte teaches the method and the computer readable medium of claims 6 and 11, further comprising the step of: mapping the at least one property to a method in a stub (column 7, lines 20-24, Apte discloses what makes a bean different from a pure object is that it has an external interface, called the properties interface, which allows a tool to read what the component is supposed to do and hook it up to other beans and plug it into another environment).

As to claims 8 and 13, Apte teaches the method and the computer readable medium of claims 6 and 11, further comprising the step of: providing a handle for the bean (column 16, lines 50-51, Apte discloses obtaining the bean as a handle; column 15, line 67 to column 16, line 2, and lines 35-38, Apte discloses the container generates a serializable handle class, providing a way to identify a bean (i.e. providing a way to identify a bean where by identifying a bean "transaction contact is performed), and the container-provided identity passes when the session bean is removed by the client at the end of the session).

As to claims 10 and 15, Apte teaches the method and the computer readable medium of claims 8 and 11, further comprising the step of: indicating if a value of one of the at least on e property has changed (column 16, lines 23-27, Apte discloses stateful session beans maintain data consistency by updating their fields each time a transaction is committed while informing of changes).

As to claim 16, Apte teaches a system for providing interaction between a client and a server, comprising:

a bean, wherein said bean further comprises:

at least one property that identifies the bean (column 7, lines 18-22, Apte discloses two java beans may be employed that implement the client object 400 and server object 402. What makes a bean different from a pure object is that it has an external interface, called the properties);

a property field that describes usage of an attribute for the bean (figure 12; column 16, lines 57-65, Apte discloses ...bean data is automatically maintained by the container using a mechanism of its choosing... when a bean chooses to have its persistence container-managed, it specifies which of its fields are to be retained);

a type that describes the bean (column 2, lines 49-51, Apte discloses the bean may have container-managed fields of both primitive and complex data-types); and

utilizing the bean as a storage entity, and transient values (column 16, lines 8-9, 53-56 and 61-65, Apte discloses an entity bean may be either a bean-managed or container-managed (i.e. where an entity bean is an entity bean is used to represent data, which can be a row or a table in a database), the bean is entirely responsible for storing and retrieving its instance data, and a bean chooses to have its persistence container-managed, and specifies which of its fields are to be retained (i.e. the fields or the values in the fields that are not retained are "transient values" since "transient values" are variables that should be excluded when objects are serialized).

Apte fails to teach explicitly a cache for storing transient values for the at least one property for subsequent retrieval after a first use.

However, Chan teaches the bean is utilized as a cache (column 7, lines 52-56, Chan discloses a type 2 access bean has a cache called a CopyHelper).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Apte in view of Chan to provide a cache for storing transient values for the at least one property for subsequent retrieval after a first use. One would be motivated to do so to allow the speed up of data transfer.

As to claim 17, Apte teaches the system of claim 16, wherein the bean maps the at least one property to a method in a stub (column 7, lines 20-24, Apte discloses what makes a bean different from a pure object is that it has an external interface, called the properties interface, which allows a tool to read what the component is supposed to do and hook it up to other beans and plug it into another environment).

As to claim 18, Apte teaches the system of claim 16, wherein the bean further comprises: a handle for the bean (column 16, lines 50-51, Apte discloses obtaining the bean as a handle; column 15, line 67 to column 16, line 2, and lines 35-38, Apte discloses the container generates a serializable handle class, providing a way to identify a bean (i.e. providing a way to identify a bean where by identifying a bean "transaction contact is performed), and the container-provided identity passes when the session bean is removed by the client at the end of the session).

As to claim 20, Apte teaches the system of claim 16, wherein the at least one property includes a value to indicate that at least one property has changed (column 16, lines 23-27, Apte discloses stateful session beans maintain data consistency by updating their fields each time a transaction is committed while informing of changes).

As to claim 21, Apte teaches the system of claim 1, wherein said means for utilizing said bean as a storage entity is configured to store transient values for said at least one property for subsequent retrieval (column 16, lines 8-9, 53-56 and 61-65, Apte discloses an entity bean may be either a bean-managed or container-managed (i.e. where an entity bean is an entity bean is used to represent data, which can be a row or a table in a database), the bean is entirely responsible for storing and retrieving its instance data, and a bean chooses to have its persistence container-managed, and specifies which of its fields are to be retained (i.e. the fields or the values in the fields that are not retained are "transient values" since "transient values" are variables that should be excluded when objects are serialized).

Apte fails to teach explicitly means for utilizing said bean as a cache entity is configured to store transient values for said at least one property for subsequent retrieval.

However, Chan teaches means for utilizing said bean as a cache (column 7, lines 52-56, Chan discloses a type 2 access bean has a cache called a CopyHelper).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Apte in view of Chan to provide means for utilizing said bean as a

cache entity is configured to store transient values for said at least one property for subsequent retrieval. One would be motivated to do so to allow the speed up of data transfer.

As to claim 22, Apte teaches the system of claim 1.

Apte fails to teach explicitly values stored in said cache entity are summarized into property sets.

However, Chan teaches values stored in said cache entity are summarized into property sets (column 5, lines 28-32, Chan discloses the access bean maintains a local cache of attributes from the enterprise bean which can be indexed (i.e. "property sets")).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Apte in view of Chan to provide values stored in said cache entity are summarized into property sets. One would be motivated to do so to allow comparing string property values.

As to claim 23, Apte teaches the system of claim 22.

Apte fails to teach property sets are configured for subsequent retrieval as one network package.

However, Chan teaches property sets are configured for subsequent retrieval as one network package (column 5, lines 24-28, Chan discloses the Access Beans increases performance when using large entity beans, in which all or a subset of their attributes persist in a persistent storage (i.e. all or a subset of their attributes are seen

as "one network package" since the "one network package" would cause performance increase)).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Apte in view of Chan to provide property sets are configured for subsequent retrieval as one network package. One would be motivated to do so to allow high reliability.

As to claim 24, Apte teaches the method of claim 6, further comprising the step of: connecting the client to an interface repository and application dictionary (figure 1).

As to claim 25, Apte teaches the method of claim 6, further comprising the step of: generating a subclass for an interface bean class (column 15, lines 59-67, Apte discloses the tool for container generates additional classes for EJB bean at deployment time, then the tool uses information it gets from the EJB to generate more class for a remote interface).

As to claim 26, Apte teaches the computer readable medium of claim 11, further comprising logic that connects to an interface repository and application dictionary (figure 1).

As to claim 27, Apte teaches the computer readable medium of claim 11, further comprising logic that generates a subclass for an interface bean class (column 15, lines

59-67, Apte discloses the tool for container generates additional classes for EJB bean at deployment time, then the tool uses information it gets from the EJB to generate more class for a remote interface).

As to claim 28, Apte teaches the system of claim 16.

Apte fails to teach explicitly transient values stored in said cache entity are summarized into property sets.

However, Chan teaches transient values stored in said cache entity are summarized into property sets (column 5, lines 28-32, Chan discloses the access bean maintains a local cache of attributes from the enterprise bean which can be indexed (i.e. "property sets")).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Apte in view of Chan to provide transient values stored in said cache entity are summarized into property sets. One would be motivated to do so to allow comparing string property values.

As to claim 29, Apte teaches the system of claim 28.

Apte fails to teach property sets are configured for subsequent retrieval as one network package.

However, Chan teaches property sets are configured for subsequent retrieval as one network package (column 5, lines 24-28, Chan discloses the Access Beans increases performance when using large entity beans, in which all or a subset of their

attributes persist in a persistent storage (i.e. all or a subset of their attributes are seen as "one network package" since the "one network package" would cause performance increase)).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Apte in view of Chan to provide property sets are configured for subsequent retrieval as one network package. One would be motivated to do so to allow high reliability.

4. Response to Arguments

Applicant's arguments with respect to claims 1, 6, 11 and 16 have been considered but are moot in view of the new ground(s) of rejection.

5. ***Conclusion***

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to El Hadji M Sall whose telephone number is 571-272-4010. The examiner can normally be reached on 8:00-4:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Ario Etienne can be reached on 571-272-4001. The fax phone number for the organization where this application or proceeding is assigned is 571-273-4010.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for

published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

El Hadji Sall
Patent Examiner
Art Unit: 2157

ARIC ETIENNE
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100